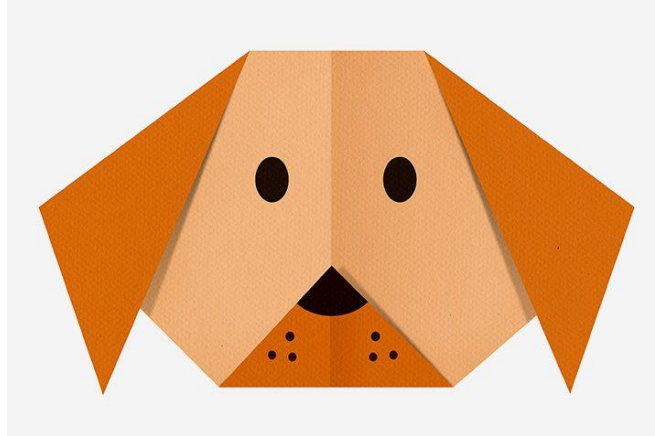


Dibujo de primitivas en 2D

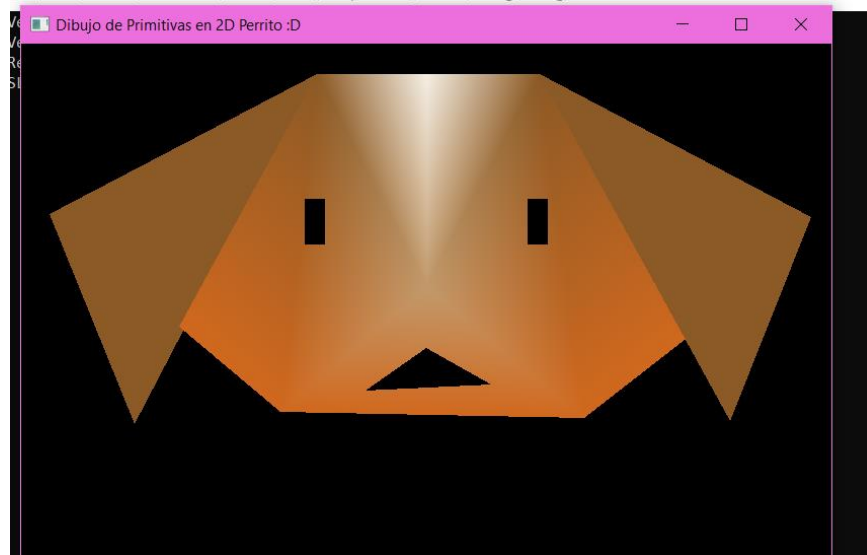
Desarrollo

Para la práctica utilice esta imagen de referencia de un perrito.



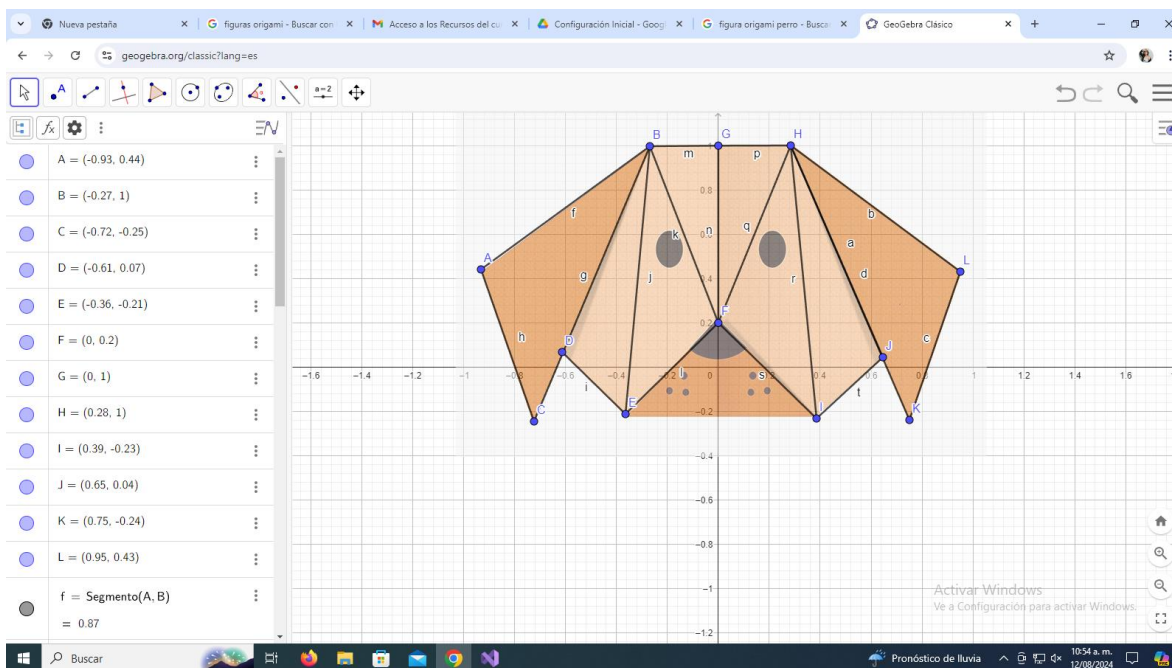
Dibujo realizado como ejercicio

Para el dibujo que realice solo la función TRIANGLE y modifique los colores para que se pudieran observar de mejor forma.



Ajustes realizados

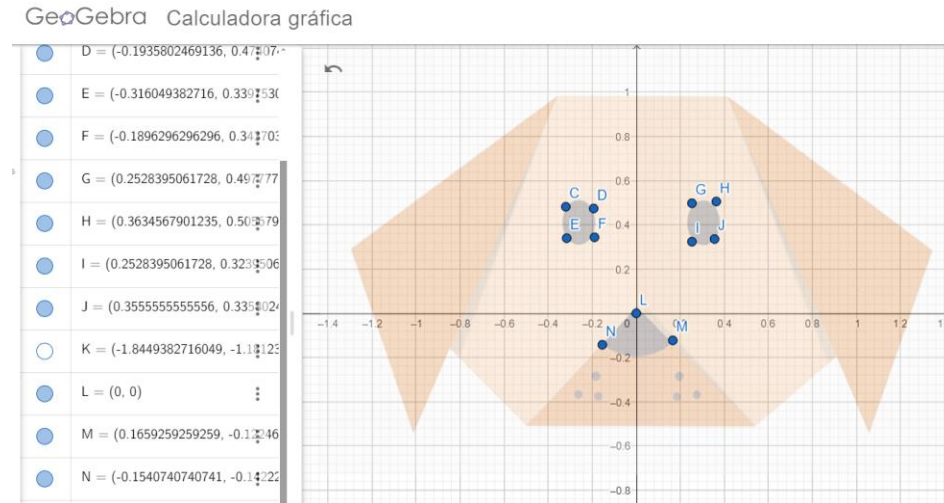
Para poder establecer los vértices de la cara del perrito, utilice la aplicación de geogebra para dividir los triángulos que se dibujarían para replicar la imagen.



Para elaborar el dibujo del perrito lo primero que hice fue agregar al código un encabezado con la X, Y y Z, las letras RGB y la letra y número del vértice para saber que coordenadas se estaban indicando y que fuese más sencillo el uso de la función.

```
//Figura perrito
// X   Y   Z       R   G   B   Vertice
-0.93f, 0.44f, 0.0f, 0.545f, 0.353f, 0.149f, //A 4
-0.27f, 0.90f, 0.0f, 0.545f, 0.353f, 0.149f, //B 5
-0.72f, -0.25f, 0.0f, 0.545f, 0.353f, 0.149f, //C 6
-0.61f, 0.07f, 0.0f, 0.82f, 0.41f, 0.12f, //D 7
-0.36f, -0.21f, 0.0f, 0.82f, 0.41f, 0.12f, //E 8
-0.0f, 0.2f, 0.0f, 0.76f, 0.60f, 0.42f, //F 9
0.0f, 0.90f, 0.0f, 0.96f, 0.94f, 0.90f, //G 10
0.28f, 0.90f, 0.0f, 0.545f, 0.353f, 0.149f, //H 11
0.39f, -0.23f, 0.0f, 0.82f, 0.41f, 0.12f, //I 12
0.65f, 0.04f, 0.0f, 0.82f, 0.41f, 0.12f, //J 13
0.75f, -0.24f, 0.0f, 0.545f, 0.353f, 0.149f, //K 14
0.95f, 0.43f, 0.0f, 0.545f, 0.353f, 0.149f, //L 15
```

Se modificó con colores en tono café y marrón para simular los de la imagen de referencia.



Para agregar sus ojos y su nariz utilice nuevos vértices que separe con comentarios para ubicarlos más fácilmente, pero siguiendo la numeración que ya estaba establecida.

```
//Ojo izq perrito
-0.30f, 0.49f, 0.0f,    0.0f, 0.0f, 0.0f, // C 16
-0.25f, 0.49f, 0.0f,    0.0f, 0.0f, 0.0f, // D 17
-0.30f, 0.34f, 0.0f,    0.0f, 0.0f, 0.0f, // E 18
-0.25f, 0.34f, 0.0f,    0.0f, 0.0f, 0.0f, // F 19

//Ojo der perrito
0.30f, 0.49f, 0.0f, 0.0f, 0.0f, 0.0f, // G 20
0.25f, 0.49f, 0.0f, 0.0f, 0.0f, 0.0f, // H 21
0.30f, 0.34f, 0.0f, 0.0f, 0.0f, 0.0f, // I 22
0.25f, 0.34f, 0.0f, 0.0f, 0.0f, 0.0f, // J 23

//Nariz perrito
0.0f, 0.0f, 0.0f,    0.0f, 0.0f, 0.0f, //A 24
0.16f, -0.12f, 0.0f, 0.0f, 0.0f, 0.0f, //B 25
-0.15f, -0.14f, 0.0f, 0.0f, 0.0f, 0.0f, //C 26
```

Para poder dibujar los triángulos utilice el arreglo de índices.

```
unsigned int indices[] = { // note that we start from 0!
    //3,2,1, // second Triangle
    //0,1,3,
    4, 5, 6,    // ABC
    5, 7, 8,    //BDE
    5, 8, 9,    //BEF
    5, 9, 10,   //BFG
    9, 10, 11,  //FGH
    9, 11, 12,  //FHI
    11, 12, 13, //HIJ
    11, 14, 15, //HKL
    8, 9, 12,   //EFI

    //Ojo izq perrito
    16, 17, 18, //CDE
    17, 18, 19, //EDF

    //Ojo der perrito
    20, 21, 22,
    21, 22, 23,

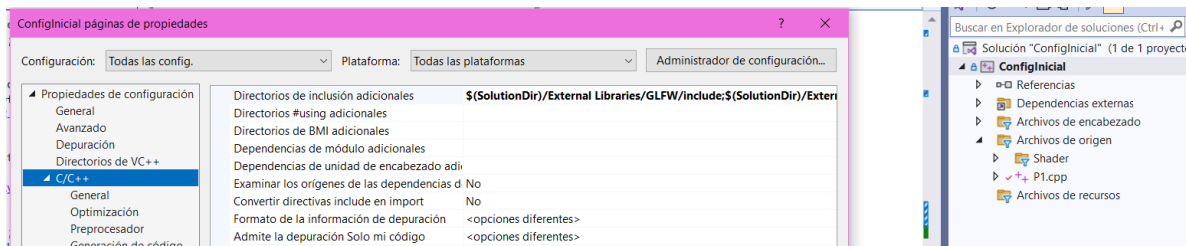
    //nariz perrito
    24, 25, 26,
};
```

Por último, para poder dibujar los triángulos desde el arreglo de índices, utilice solo una función para mandar a llamar los vértices.

```
glDrawElements(GL_TRIANGLES, 42, GL_UNSIGNED_INT, 0);
```

Ajustes extra.

Para que el proyecto se pueda configurar correctamente, es necesario primero agregar un archivo .cpp para que el visual nos permita agregar librerías en la parte de propiedades.



Liga de GitHub:

<https://github.com/bereenicee7/Curso Computacion Grafica.git>